Calculate a Cumulative Average in Python?

Authored by

stats writer

November 30, 2025

To calculate a cumulative average in Python, you need to create a loop that will iterate through a list of numbers and add each number to the average. The cumulative average is then calculated by dividing the sum of all the numbers in the list by the total number of elements in the list. This will give you the cumulative average of the list of numbers.

A **cumulative average** tells us the average of a series of values up to a certain point.

You can use the following syntax to calculate the cumulative average of values in a column of a pandas DataFrame:

df.expanding().mean()

The following example shows how to use this syntax in practice.

Example: Calculate Cumulative Average in Python

Suppose we have the following pandas DataFrame that shows the total sales made by some store during 16 consecutive days:

import pandas as pd
import numpy as np

#create DataFrame
df = pd.DataFrame({'day': ,
'sales': })

#view first five rows of DataFrame
df.head()

day sales
0 1 3
1 2 6
2 3 0
3 4 2
4 5 4

We can use the following syntax to calculate the cumulative average of the sales column:

#calculate average of 'sales' column
df.expanding().mean()

0 3.000000
1 4.500000
2 3.000000
3 2.750000
4 3.000000
5 2.666667
6 2.285714
7 2.125000
8 2.333333
9 2.800000
10 2.818182
11 2.833333
12 3.230769
13 3.214286
14 3.333333
15 3.437500
Name: sales, dtype: float64

We would interpret the cumulative average values as:

The cumulative average of the first sales value is **3**.
The cumulative average of the first two sales values is **4.5**.
The cumulative average of the first three sales values is **3**.
The cumulative average of the first four sales values is **2.75**.

And so on.

Note that you can also use the following code to add the cumulative average sales values as a new column in the DataFrame:

#add cumulative average sales as new column
df = df.expanding().mean()

#view updated DataFrame
df

day sales cum_avg_sales
0 1 3 3.000000
1 2 6 4.500000
2 3 0 3.000000
3 4 2 2.750000

4 5 4 3.000000
5 6 1 2.666667
6 7 0 2.285714
7 8 1 2.125000
8 9 4 2.333333
9 10 7 2.800000
10 11 3 2.818182
11 12 3 2.833333
12 13 8 3.230769
13 14 3 3.214286
14 15 5 3.333333
15 16 5 3.437500

The **cum_avg_sales** column shows the cumulative average of the values in the "sales" column.

The following tutorials explain how to calculate other common metrics in Python: